三维灰体辐射传递系数蒙特卡洛的 GPU 计算 *

刘　硕，朱希安，王占刚，厉夫兵

(北京信息科技大学 信息与通信工程学院，北京 100101)

摘　要： 为了计算漫反射灰体表面之间的辐射传递系数，建立了三维灰立方体模型，在所建立的三维立体空间内表面选取发射表面，利用蒙特卡洛方法生成发射光线，并对光线进行光线追踪，计算发射表面与其他表面之间的辐射传递系数。针对基于蒙特卡洛法计算三维灰体表面之间辐射传递系数时耗时过长的问题，进行了基于显卡 GPU 实现蒙特卡洛并行计算的研究。利用统一编程架构 CUDA 对蒙特卡洛法进行了编程实现，采用 CUDA 随机数发生算法并行处理了求解辐射传递系数中的光线追踪过程，实现了计算辐射传递系数的并行方案。实验结果表明，相对于 CPU 串行计算方法，基于 CUDA 架构的并行蒙特卡洛法求解辐射传递系数可以取得高达 80 倍的计算加速比。

关键词： 蒙特卡洛；辐射传递系数；GPU；并行计算；三维灰体

中图分类号： TP391.7　　doi: 10.3969/j.issn.1001-3695.2018.05.0358

Monte Carlo for calculating radiative heat transfer coefficient of 3D gray body based on GPU

Liu Shuo, Zhu Xian, Wang Zhangang, Li Fubing

(*School of Information & Communication Engineering*, *Beijing Information Science & Technology University*, *Beijing* 100101, *China*)

Abstract: In order to calculate the radiative heat transfer coefficient between the diffuse surface of the 3D gray body, a cell cube model was setup to represent the gray surface, rays emitted from one surface (i. e. , emission surface) are generated by using Monte Carlo method. Multiple reflections for each ray were being traced until it is totally absorbed. Energy absorbed at each ray-facet intersection is recorded to calculate the heat transfer coefficient between the emission surface and any other surface. To solve the over-consumption issue of radiative heat transfer coefficient based on Monte Carlo method between the diffuse surface of the 3D gray body, a Monte Carlo method based on GPU has been used to realize the parallel computation in this paper. Also in this paper, the unified programming architecture (CUDA) has been used to get the realization of programming the Monte Carlo method, according to the CUDA random number generation algorithm, the tracing process of the radiative transfer coefficient has been parallel computed, and a parallel scheme for calculating the radiative transfer coefficient has been realized. The experimental results show that, compared with the CPU serial calculation method, the parallel Monte Carlo method based on CUDA architecture can obtain the calculated acceleration ratio of 80 times as high as the radiative heat transfer coefficient.

Key words: monte carlo; radiative heat transfer coefficient; parallel computing ; GPU; 3D gray body

0　引言

众所周知，不同温度的物体之间的热量传递主要通过导热、对流和热辐射三种方式来实现，尤其在真空中，唯一可以依赖的换热方式就是辐射换热[1]。正因为如此，近三四十年来随着火箭技术、人造卫星、宇宙飞船和核工业的发展，辐射换热的理论与实验研究受到了人们的极大重视，根据红外物理理论：物体之间辐射换热的研究最重要的是物体之间的辐射传递系数的计算。蒙特卡洛方法是一种通过随机变量的统计实验来求解数学物理或工程技术问题的数值方法，已有上百年的历史，但早期由于模拟实验工具的限制，真正用于解决实际问题还比较少[2]。近二十多年来，在实际应用中，随着计算机的高速发展，为红外辐射特性计算的三维建模方面提供了有利条件。与此同时，蒙特卡洛法在现实的应用中随着计算机的高速发展，被逐步用于辐射换热问题的计算。

蒙特卡洛法求解表面辐射换热近年来得到了广泛的应用。

收稿日期： 2018-05-22；**修回日期：** 2018-09-06　**基金项目：** 北京市科技创新服务能力建设-基本科研业务费（科研类）(71E1810969)；北京市科技创新服务能力建设-提升计划项目（PXM2017_014224_000009）

作者简介： 刘硕（1992-），男，河北省邯郸市磁县人，硕士研究生，主要研究方向为 GPU 高性能计算（765375297@qq.com）；朱希安（1962--），男，教授，博士，主要研究方向为智慧感知与信息处理、虚拟现实等；王占刚（1977 年-），男，博士后，副教授，主要研究方向为三维建模与可视化研究；厉夫兵（1982-），男，博士后，讲师，主要研究方向为红外辐射特性建模、光线追踪算法等.

文献[3,4]介绍了在辐射换热领域，蒙特卡洛法求解辐射换热是基于概率统计理论的随机抽样方法，利用重复的统计实验对所研究物理或数学问题进行直接或间接地描述，真实反映事物特点和物理实验过程。文献[5]分析了在表面热辐射问题中蒙特卡洛法求解辐射换热结果的精确度问题，在统计实验特别大时，蒙特卡洛法的结果误差更小，给出了蒙特卡洛法是求解辐射换热数值的金标准。文献[6]中根据辐射换热中的公式法，对求解过程分析总结后，提出了漫反射辐射界面特性的单元表面之间辐射传递系数的新型计算方法，并且与蒙特卡洛法计算结果进行分析对比，间接证明了蒙特卡洛法的精确度和准确性，为灰体表面之间的辐射传递系数的研究奠定了结果标准。

随着计算机技术的发展，高性能计算速度从每秒百万次浮点运算提高到了每秒千万亿次浮点运算，在三维建模仿真、软硬件技术、算法实现和计算机应用等方面都得到了巨大的发展。充分利用 GPU 硬件资源对三维模型建立和算法进行并行通用计算研究是一个热门的研究方向，所以近年来很多学者利用 GPU 实现各种算法和三维模型仿真。文献[7，8]提出了蒙特卡洛法可并行化进行分析和优化，根据串行和并行对比数值实验结果，对蒙特卡洛法的并行结构和理论提出了实现方案。文献[9~11]进行了蒙特卡洛法在 GPU 上加速的分析和研究，在高性能计算领域搭建了蒙特卡洛法框架，实现了蒙特卡洛法的加速计算。文献[12，13]可以发现在 GPU 架构下，提出了蒙特卡洛算法设计和对算法设计进行了优化分析，但是没有实现蒙特卡洛法求解辐射传递系数。

本文是为了利用蒙特卡洛法求解三维单元灰立方体内表面之间的辐射传递系数，建立了三维红外辐射特性模型，并且利用 CUDA 平台，实现求解辐射传递系数的蒙特卡洛法 GPU 加速，分析了 CUDA 平台上求解辐射传递系数的效果，将其得到的效果与 CPU 实现蒙特卡洛法计算结果进行对比分析；利用文献[6]中新型的求解辐射传递系数公式法所得结果，与本文 CUDA 框架下所得结果进行准确性分析；最后本文对 GPU 加速计算效果进行了研究。辐射传递系数应用于计算三维物体表面的温度分布，为下一步三维空间红外辐射特性建模、计算三维空间物体内的温度分布和辐射换热的精确、快速计算奠定重要的参数条件。

1 蒙特卡洛法计算辐射传递系数

蒙特卡洛法是一种随机模拟和统计的实验方法，本文利用蒙特卡洛法来研究三维灰立方体内表面之间的辐射传递系数，光线在灰体表面发生漫反射，如图 1 所示，漫反射首先要确定发射表面光束发射的位置和发射的方向：在设定的发射表面 M_1 上根据式（1）随机选取一个发射点 A。

$$\begin{cases} x = R_x \\ y = R_y \\ z = 0 \end{cases} \tag{1}$$

其中： R_x、R_y 是在[0,1]区间满足均匀分布的随机变量。

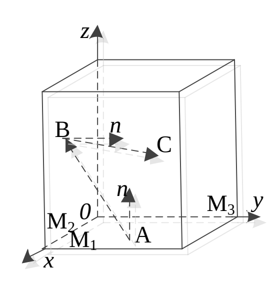


图 1 三维灰立方体内表面漫反射示意图

在三维立体的表面上确定了发射点后，在发射点根据蒙特卡洛法来确定随机的发射方向：在发射点所在的表面上，利用该表面的法线 n 所在的方向作为 z 轴正方向来建立该表面的本体坐标系，在该表面的本体坐标系中，由式(2)可以求得随机发射方向。

$$\begin{cases} \theta_f = \operatorname{asin}\sqrt{R_\theta} \\ \varphi_f = 2\pi R_\varphi \end{cases} \tag{2}$$

其中：θ_f、φ_f 分别为发射光束的天顶角和方位角，R_θ、$R\varphi$ 为[0,1]区间内均匀分布的随机数。

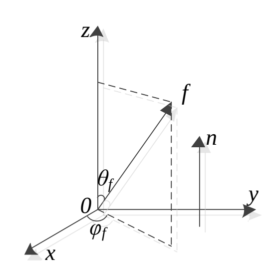


图 2 本体坐标系中的发射方向

在空间三维立方体的任意六个表面都存在六个不同的本体坐标系如图 2 所示，因此在立方体的六个内表面的本体坐标系中，利用坐标转换矩阵[14]，将本体坐标系所在的三维空间半球内的发射方向转换到参考坐标系中表示，便于光线追踪计算。在计算三维灰体表面的辐射传递系数时，表面向所在半球空间发射的能量被视为根据公式发射的光线的能量和。假定发射表面发出光线总数为 N，每条光线所携带的能量为

$$e = \frac{(1-\varepsilon)\sigma T^4}{N} \tag{3}$$

其中：T 为表面的瞬态温度，ε 为表面的反射率，σ 为玻尔兹曼常数。

在灰体表面，反射光线所携带的能量依表面反射率 ε 而衰减的，如图 1 所示，初始光线从 A 发出所携带能量为 e，第一次到达表面 M_2 后反射光线的能量变为 εe，表面 M_2 所吸收的能量为（1-ε）e。然后在该表面根据上文的蒙特卡洛漫反射原理确定反射光线后，到达 M_3 进行第二次漫反射，经过第二次漫反射后，反射能量变为 $\varepsilon^2 e$，M_3 所吸收的能量为（1-ε）εe。以此类推，第 k 次漫反射的反射光线能量为 $\varepsilon^k e$。给定阈值 η，当反射光线的能量 $\varepsilon^k e$ 与 e 之比低于阈值 η 即可视为光线的能量完全被立方体表面吸收。遍历 N 条发射光线，统计每个反射点所

在的表面的能量，可求出发射表面对其他表面的辐射传递系数为：

$$F_{1-i} = \frac{E_i}{Ne} \tag{4}$$

其中：E_i 为发射表面 1(即 M_1)的辐射能量中被表面 i 吸收的部分，包括表面 1 对表面 i 直接射入的能量和被其他表面反射后到达 i 的能量。

2　蒙特卡洛法的 CUDA 实现

2.1　CUDA 实现蒙特卡洛

CUDA 是 NVIDIA 公司开发的并行计算框架。开发者通过 CUDA 提供的虚拟指令可以使用并行计算元素和 GPU 内存并使用。有了 CUDA 的支持，开发者可以像使用 CPU 一样方便使用 NVIDIA 公司的 GPU。CUDA 采用多层次的存储器结构，并用 C 语言作为程序语言，如图 3 所示 CUDA 架构模型，编程人员通过写简单的 GPU 内核程序，实现并行计算的功能。一个内核程序可以被多个线程并行执行。多个线程可以组织一个线程块，线程块内的线程共享内存和寄存器等资源。多个线程块组成一个内核网格，内核网格内的线程块间相互独立地并行执行。

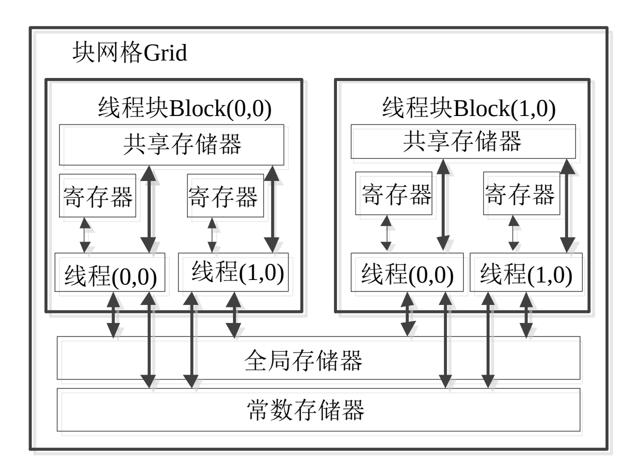


图 3　CUDA 架构模型

基于 CUDA 的程序计算基本步骤为：首先，分别在 GPU 和 CPU 上分配内存，在 Host 端(CPU)初始化相关数据；其次，设置 grid 和 block 的大小，在 Device 端(GPU)进行并行计算；最后，当计算结果满足终止准则后，将数据拷贝到 Host 端，得到最终结果，释放 GPU 和 CPU 的内存。本文根据 CUDA 程序计算的步骤，对蒙特卡洛法进行实现的流程如图 4 所示。首先在 CPU 端对三维灰体进行建模，根据红外辐射特性，定义三维灰体的参数，设置灰体表面的辐射特性参数和光线总数；然后将在 CPU 中定义大量的数据然后传送到 GPU 后，在 GPU 线程中实现蒙特卡洛法的红外辐射光线的一系列追踪处理：发射、漫反射、求交、光线追踪、阈值判断和能量计算等，计算数据并保存三维灰体每个红外辐射灰体表面所吸收光线的能量；最后，当光线总数满足定义要求后，将数据结果拷贝至 CPU 中，根据式(4)得到灰体表面之间的辐射传递系数。

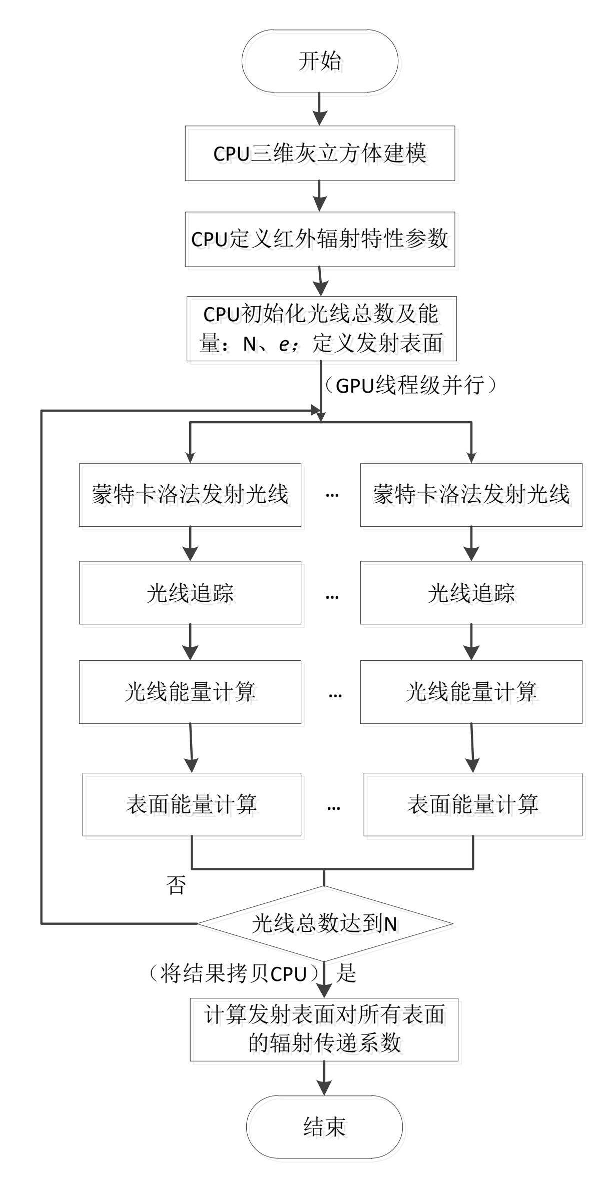


图 4　CUDA 实现蒙特卡洛流程图

2.2　CUDA 随机数发生器

本文通过蒙特卡洛方法，计算发射表面对其他表面的辐射传递系数，每条发射光线的位置和方向、反射光线的方向都由计算机随机产生，这就需要一个高效的随机数发生器，而使用该随机数的程序段运行于 CUDA 设备上，结合求解辐射传递系数对随机数均匀性的要求，如何在 CUDA 设备中产生快速产生随机数是蒙特卡洛法的实现的关键。

根据基本的数学常识：常用的随机数概率模型有三种，即均匀分布、正态分布和指数分布模型。在众多的随机数分布模型中，服从均匀分布的随机数是基础，其他概率分布的模型都可以通过均匀分布的随机数进行一定的函数转换而得到[15]。根据图 4 蒙特卡洛流程图可以发现：在三维红外辐射特性建模后，需要在 CPU 中产生大量的随机数然后传送到 GPU 中，但是采用这种方式来处理这部分随机数时，CPU 与 GPU 之间的数据传输和读取操作的时间开销代价太大。

因此，本文采用了平方取中算法来产生均匀随机分布随机数。平方取中法对设置的随机数种子进行平方，去掉开始的几位，再去掉结束的几位，剩下的即为需要的随机数。并将其作

为随机数种子再次进行同样的计算，如此反复，就得到了一系列的随机数序列，再对其进行归一化处理，即转换为[0,1]区间内的标准浮点数。本文为实现 CUDA 设备中产生随机数，把线程号、循环次数和当前时间的乘积作为随机数种子，多个线程同时产生随机数，这样所有的线程同时产生的随机数服从均匀分布。

2.3 光线追踪

光线能量传输是蒙特卡洛方法的基础，单条光线的追踪是利用蒙特卡洛方法对辐射换热问题进行求解的基本单元。根据蒙特卡洛法求解辐射传递系数的原理，如图 4 所示，在初始化参数后，在 GPU 线程并行实现光线的追踪。

结合以上蒙特卡洛法原理与 CUDA 实现方法，如图 5 所示，本文基于 CUDA 的光线追踪并行程序实现的详细算法：

a)在 GPU 端对变量参数进行内存分配空间，利用 CUDA 随机数发生器在设定的发射表面生成随机发射点和发射方向，根据基本的数学计算，求解出由发射点和发射方向所确定的射线。

b)求解该射线与三维立方体的交点，计算该交点所在表面吸收的能量和光线剩余的能量。若光线剩余的能量低于表面吸收阈值，光线追踪结束。若光线剩余能量不低于表面吸收阈值，在交点处利用 CUDA 随机数发生器生成随机反射方向。

c)由交点和随机反射方向确定反射射线后，重复 b)过程，直至光线追踪结束。

d)将所计算的表面吸收的能量从 GPU 内存复制到主存中，释放 GPU 内存空间。

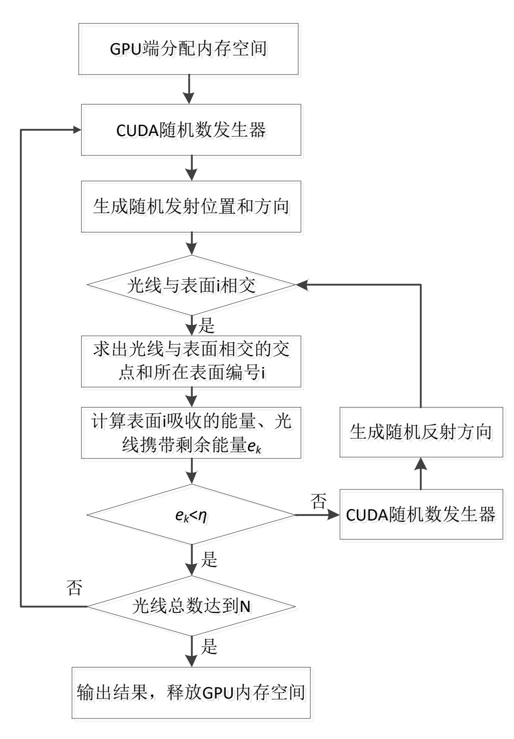


图 5 蒙特卡洛光线追踪流程图

3 实验结果与分析

3.1 参数设置与实验环境

本文的实验参数设置如下：

三维灰体建模为：灰体立方体的长、宽、高都设为 1(长度单位归一化)；根据所建立的灰体立方体设置空间坐标系，x 轴、y 轴和 z 轴的单位长度按照立方体的归一单位进行划分；三维立体内表面反射率 ε=0.8；内表面吸收阈值 η=0.01；在空间坐标系中，三维立体内表面的编号根据内表面平面所在的表面方程 z=0、y=0、x=0、y=1、x=1、z=1 依次编号为 1-6；为了方便研究以及分析实验结果，本文设置表面 1 号即平面方程为 z=0 的表面为光线发射表面。三维灰体内表面反射率和光线发射总数 N 根据实验内容要求设定；光线初始携带能量 e=1。

本文实验的硬件平台 PC 机，2.60GHz Intel E5-2697 v3 CPU，256GB 内存和 NVIDIA Tesla K40c GPU。软件环境是 Windows 7，Visual studio 2013，CUDA 8.0。

3.2 实验结果分析

设定发射光线总数 $N=10^6$ 时，为了保证结果的可靠性，本文将程序在 GPU 和 CPU 上分别运行 50 次，取实验结果的平均值作为最终结果，如表 1 和 2 所示。

表 1 CPU 计算辐射传递系数结果

表面号	辐射传递系数
1	0.045904
2	0.191006
3	0.184511
4	0.191192
5	0.190510
6	0.190361

表 2 GPU 计算辐射传递系数结果

表面号	辐射传递系数
1	0.038394
2	0.192314
3	0.192216
4	0.192268
5	0.192118
6	0.192407

表 3 文献[6]新型公式法所得结果

表面号	辐射传递系数
1	0.38461
2	0.192304
3	0.192306
4	0.192310
5	0.192313
6	0.192305

对比 CPU 和 GPU 计算结果表明：表 3 为文献利用辐射能量传递过程的分解求解出的辐射传递系数。对比表 1、表 2 和表 3 可知：若光线数量足够大，利用蒙特卡洛法计算辐射传递系数和文献中按解方程所得结果数据基本一致，间接证明了文献介绍蒙特卡洛法是计算灰体表面辐射传递系数的金标准，同时，基于 GPU 实现蒙特卡洛计算辐射传递系数与 CPU 实现蒙

特卡洛计算具有相同的结果，因此，GPU 完全可以代替 CPU 进行基于蒙特卡洛法计算灰体表面之间的辐射传递系数。

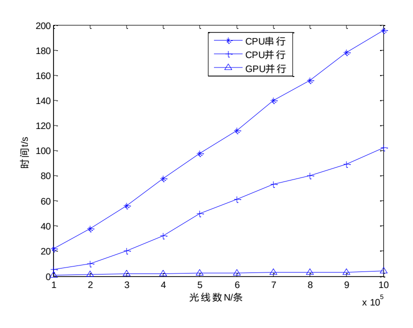


图 6　CPU 串行、CPU 并行和 GPU 并行程序运行时间随光线数变化图

分别设置发射光线总数为 10^5、$2×10^5$、$3×10^5$、$4×10^5$、$5×10^5$、$6×10^5$、$7×10^5$、$8×10^5$、$9×10^5$ 和 10^6，CPU 串行、CPU 并行和 GPU 并行对蒙特卡洛法求解辐射传递系数求解所用时间如图 6 所示。经过对比结果表明：相比于 CPU 串行高达数百秒的计算时间，虽然利用 CPU 多核并行处理蒙特卡洛法的运行时间能加速到百秒内，取得了一定的加速效果。但是采用 GPU 进行并行计算时，计算时间一直在 5s 内，相比 CPU 多核并行处理，更加提高了加速率，极大地提高了计算效率。

对于并行加速的指标评判，本文采用加速比作为 CUDA 求解辐射传递系数程序的并行加速性能评价指标。加速比是指同一个任务在串行处理系统和并行处理系统中所耗费的时间的比率，是用来衡量并行系统或程序并行化的重要指标[17]。其计算方式为

$$P = \frac{T_S}{T_M} \tag{5}$$

其中：P 是加速比，T_s 是串行处理耗时，T_m 是在并行系统中的运行时间。

表 4　GPU 随光线数计算辐射传递系数加速比

光线数	加速比
200000	18.6
400000	23.806
600000	48.5774
800000	69.11692
1000000	80.22980

根据图 6 结果，分别设置发射光线总数为 $2×10^5$、$4×10^5$、$6×10^5$、$8×10^5$、和 10^6 时，利用公式计算加速比结果如表 4 所示，结果表明：利用 CUDA 实现蒙特卡洛法计算灰体表面之间的辐射传递系数的时间远远小于 CPU 串行和 CPU 并行计算所用时间，当光线数为 100 万时，GPU 的加速比高达 80；具有金标准之称的蒙特卡洛法的精确度随光线数的增加而提高，所以利用 GPU 加速后的蒙特卡洛法，光线数越大，精确度越高，同时加速比也随之增加，因此利用 CUDA 实现蒙特卡洛法计算灰体表面之间的辐射传递系数极大的提高了求解时间和速度，并且加上金标准之称，为求解三维物体表面之间的辐射换热奠定了精确快速的计算参数。

4　结束语

随着三维红外建模技术和高性能计算技术的爆炸式发展，大量数据的处理对实时性的要求越来越高，基于 CUDA 的数据并行处理作为一种有效提高数据处理速度，提高应用系统响应实时性的重要技术手段得到了广泛应用。在辐射传热学中，三维红外辐射特性建模一直是国内外研究热点，蒙特卡洛法是求解灰体表面辐射传递系数的金标准，辐射传递系数是计算不同温度灰体表面之间传递能量和辐射换热的重要参数，是影响研究三维物体红外辐射特性的关键因素。

本文根据红外辐射特性最新的研究发展需要，建立了三维空间灰体的红外辐射特性模型，提出了用 GPU 加速蒙特卡洛法的架构和流程，实现了基于 CUDA 并行加速蒙特卡洛法计算三维灰体表面之间的辐射传递系数。结果表明：利用 GPU 加速后的蒙特卡洛法计算辐射传递系数可以获得特别可观的加速比，极大地提高了求解效率，有助于未来三维红外辐射特性建模研究，为三维空间物体的辐射换热的研究披上了高性能计算的外衣，开辟了红外物理的辐射特性计算加速基础。

参考文献：

[1] 卞伯绘. 辐射换热的分析与计算 [M]. 北京: 清华大学出版社, 1988: 81-83. (Bian Bohui. Analysis and calculation of radiation heat transfer [M]. Beijing: Tsinghua University Press, 1988: 81-83.)

[2] 梁英, 肖卫国, 王力, 等. 基于可达辐射能的目标表面辐射传递系数计算方法 [J]. 红外, 2014, 35 (2): 44-48. (Liang Ying, Xiao Weiguo, Wang Li, *et al.* Calculation method of radiative heat transfer coefficient of target surface based on reachable radiation energy [J]. Infrared, 2014, 35 (2): 44-48.)

[3] 李亚男, 刘鑫, 刘佳琪, 等. 蜂窝结构瞬态传热性能仿真研究 [J]. 导弹与航天运载技术, 2015 (1): 88-91. (Li Yanan, Liu Xin, Liu Jiaqi, *et al.* Simulation study on transient heat transfer performance of honeycomb structure [J]. Missile and Spaceflight Transport Technology, 2015 (1): 88-91.)

[4] 厉夫兵, 冷俊敏, 李红莲. 诱饵球镜面反射内壁的辐射传递系数计算 [J]. 现代防御技术, 2017, 45 (3): 193-199+214. (Li Fubing, Leng Junmin, Li Honglian. Calculation of radiative transfer coefficient of reflector inside the bait ball [J]. Modern Defense Technology, 2017, 45 (3): 193-199+214.)

[5] 汪洪源, 陈赟. 天基空间目标红外动态辐射特性建模与仿真 [J]. 红外与激光工程, 2016, 45 (5): 19-25. (Wang Hongyuan, Chen Yun. Modeling and simulation of infrared dynamic characteristics of space-based targets [J]. Infrared and Laser Engineering, 2016, 45 (5): 19-25.)

[6] 张伟清, 宣益民, 韩玉阁. 单元表面间辐射传递系数的新型计算方法 [J]. 宇航学报, 2005, 26 (1): 77-80, 85. (Zhang Weiqing, Xuan Yimin, Han Yuge. A new method of radiative transfer coefficient between unit surfaces [J] , Journal of Astronautics, 2005, 26 (1): 77-80, 85.)

[7] 张永强, 徐宗昌, 孙寒冰, 等. 基于蒙特卡洛仿真和并行粒子群优化算法的携行备件优化 [J]. 兵工学报, 2016, 37 (1): 122-130. (Zhang Yongqiang, Xu Zongchang, Sun Hanbing, *et al.* Optimization of spare parts based on Monte Carlo simulation and parallel particle swarm optimization algorithm [J]. Acta Armamentarii, 2016, 37 (1): 122-130.)

[8] Tang Ming, Zhao Jieyi, Tong Ruofeng, *et al.* GPU accelerated convex hull computation [J]. Computers & Graphics, 2012, 36 (5): 498-506.

[9] 王克, 薛小超, 朱朋海. 非线性方程的多核并行蒙特卡洛求解方法 [J]. 现代计算机, 2014 (20): 38-44. (Wang Ke, Xue Xiaochao, Zhu Penghai. Multi core parallel Monte Carlo method for solving nonlinear equations [J]. Modern Computer, 2014 (20): 38-44.)

[10] 李孟伟. GPU-CPU 混合架构下燃烧室辐射换热求解器的设计与实现 [D]. 绵阳：西南科技大学, 2016. (Li Mengwei. Design and implementation of combustor radiant heat exchanger solver based on GPU-CPU hybrid structure [D]. Mianyang: Southwest University of Science & Technology, 2016.)

[11] Zhu Dingqiang, Shen Wentao, Cai Guobiao. Numerical simulation and experimental study of factors influencing the optical characteristics of a spatial target [J]. Applied Thermal Engineering. 2013, 50: 749-762.

[12] 陈杰, 吴晓迪, 杨星, 等. 基于蒙特卡洛改进算法的空间热流计算 [J]. 红外技术, 2017, 39 (1): 91-94. (Chen Jie, Wu Xiaodi, Yang Xing, *et al.* Calculation of space heat flow based on Monte Carlo improved algorithm [J]. Infrared Technology, 2017, 39 (1): 91-94.)

[13] Devi J, Kruis F E. A fast Monte Carlo GPU based algorithm for particle breakage [C]// Proc of International Conference on Control, Decision and Information Technologies. 2017: 0784-0789.

[14] Yusoff M N S, Jaafar M S. Performance of CUDA GPU in Monte Carlo simulation of light-skin diffuse reflectance spectra [C]// Biomedical Engineering and Sciences. 2013: 264-269.

[15] 厉夫兵, 李红莲. 灰体球内表面的辐射传递系数 [J]. 湘潭大学自然科学学报, 2017, 39 (2): 86-89. (Li Fubing, Li Honglian. The radiative transfer coefficient of the inner surface of the ash ball [J]. Natural Science Journal of Xiangtan University, 2017, 39 (2): 86-89.)

[16] 刘欢, 刘志勤, 李凌, 等. 一种基于 CUDA 平台的随机数算法研究与实现 [J]. 计算机应用研究, 2017, 34 (9): 2727-2731. (Liu Huan, Liu Zhiqin, Li Ling, *et al.* Research and implementation of random number algorithm based on CUDA platform [J]. Application Research of Computers, 2017, 34 (9): 2727-2731.)

[17] Tang Ming, Zhao Jieyi, Tong Ruofeng, *et al.* SMI 2012: full GPU accelerated convex hull computation [J]. Computers & Graphics, 2012, 36 (5): 498-506.